

# TECHNICAL REPORTS

(NASA-CR-192755) APPLICATIONS OF  
NEURAL NETWORKS IN ROBOTICS AND  
AUTOMATION FOR MANUFACTURING  
(Rensselaer Polytechnic Inst.)  
23 p

N93-71636

Unclass

29/63 0153767

*6/10/87*  
*7/11/87-CR*  
*153767*  
*P. 23*



Center for Intelligent  
Robotic Systems  
for Space Exploration

Rensselaer Polytechnic Institute  
Troy, New York 12180-3590

Technical Reports  
Engineering and Physical Sciences Library  
University of Maryland  
College Park, Maryland 20742

**APPLICATIONS OF NEURAL NETWORKS IN ROBOTICS AND  
AUTOMATION FOR MANUFACTURING**

**by**

**Professor Arthur C. Sanderson  
Professor & Chairman  
Electrical, Computer, & Systems Engineering Department  
Rensselaer Polytechnic Institute  
Troy, New York 12180**

**A Summary of Remarks Presented at the NSF Workshop on  
Neural Networks & Robotics**

**Portsmouth, New Hampshire**

**October 1988**

**APPLICATIONS OF NEURAL  
NETWORKS IN ROBOTICS AND  
AUTOMATION FOR  
MANUFACTURING**

**By:**

**A.C. Sanderson**

**Department of Electrical, Computer and Systems Engineering  
Department of Mechanical Engineering, Aeronautical  
Engineering & Mechanics  
Rensselaer Polytechnic Institute  
Troy, New York 12180-3590**

**October 1988**

**CIRSSE Document #30**

## ABSTRACT

Neural networks provide an important approach to adaptive and learning behavior in robotics and automation systems for manufacturing applications. Computational neural networks with capabilities for supervised learning, matching, and generalization offer an efficient means for implementation of new automation systems by providing tools which facilitate the integration of sensors and mechanisms, the adaptation of control structures to new situations, the flexible planning and scheduling of tasks and tasks sequences, and the increase in reliability through adaptive learning of actions. Automated assembly is one example of a manufacturing task which requires extensive integration of mechanisms and sensors. New computational approaches to geometric reasoning, task planning, motion planning, flexible sensor-based control, and error recovery would decrease the implementation cost and increase the reliability of these systems. Robotics often requires an experimental approach to the development and demonstration of new techniques, and the effective use of neural networks in a specific applications domain such as manufacturing will require consideration of the problems and constraints posed by that domain.

## 1. INTRODUCTION

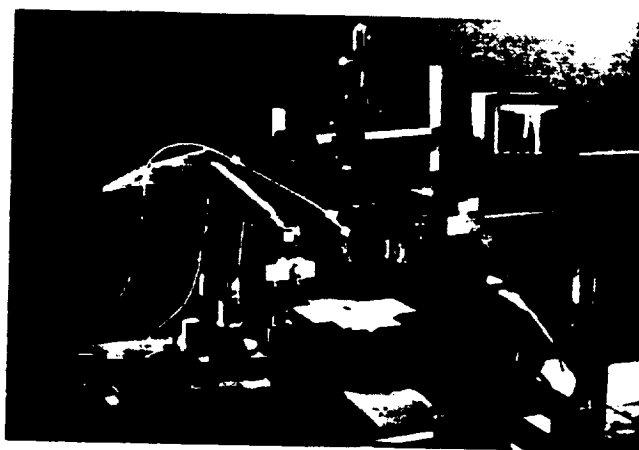
This paper provides an overview of applications of neural networks in robotics and automation with particular emphasis on potential applications to manufacturing. The paper summarizes some individual views presented at the NSF Workshop and provides examples from the author's work. It does not attempt to review the literature in these fields. Background on the basic technologies may be found in standard texts such as [1]. The paper will focus on issues which arise in those applications that lend themselves to solutions by techniques involving adaptive or learning systems such as neural networks.

Neural network computing methods provide one approach to the development of adaptive and learning behavior in robotic systems for manufacturing. Computational neural networks have been demonstrated which exhibit capabilities for supervised learning, matching, and generalization for problems on an experimental scale. In this paper we point to a number of issues in the manufacturing applications of robotics where these capabilities will be extremely important. Supervised learning could improve the efficiency of training and development of robotic systems. Matching provides a means to execute the learned behavior and will be important in areas such as industrial inspection and control and task execution functions. Generalization capabilities of neural networks will require more long-term research, but could facilitate the flexibility of systems in their capacity to adapt to new tasks. Several examples of these applications are discussed in this paper.

Manufacturing continues to be economically the most important application of robotics and automation technology. The use of adaptive and learning capabilities in automation systems to simplify the implementation process and to improve the reliability of these systems, may have a tremendous practical impact. Robotics and automation technology has an important role in a variety of different manufacturing tasks. These include such areas as parts handling, metal cutting, paint spraying, plastic molding, welding, fastening systems, assembly, and many other more specialized operations. In manufacturing, most such applications are integrated into a larger manual or automated system, and the requirements of this overall systems function often dominate the success and capability of any particular operation. Therefore, it is important in evaluating a particular technology or the potential impact of a new technology such as neural networks to assess the impact on the overall system. As a means to describe some characteristics of these manufacturing systems, in the next section we will consider the example of automated assembly systems in more detail. In Section 3 we will discuss overall issues and opportunities for neural networks in manufacturing applications.

## 2. AUTOMATED ASSEMBLY: AN EXAMPLE

The key obstacle to making manufacturing systems work economically and efficiently in industry today is most often the overall systems coordination and not the control of specific devices. An example of a robotic assembly work cell is shown in the photograph in Figure 1 [2]. This work cell consists of three robot arms, a movable work surface and fixturing, and several different types of sensors. The function of this assembly work cell is to acquire parts that are presented to the system, orient the parts in a prescribed manner, mate the parts into predetermined relationships, and fasten the parts into a final stable configuration. While such an assembly work cell depends on the speed and accuracy of its individual components, the overall capabilities are most closely related to its capacity as a system to reliably integrate functions of positioning, grasping, and sensing. Currently, the difficulty in developing such systems for manufacturing applications is in the implementation, planning, programming, and coordination of the various devices in order to create a reliable system, rather than in the choice of particular mechanisms.



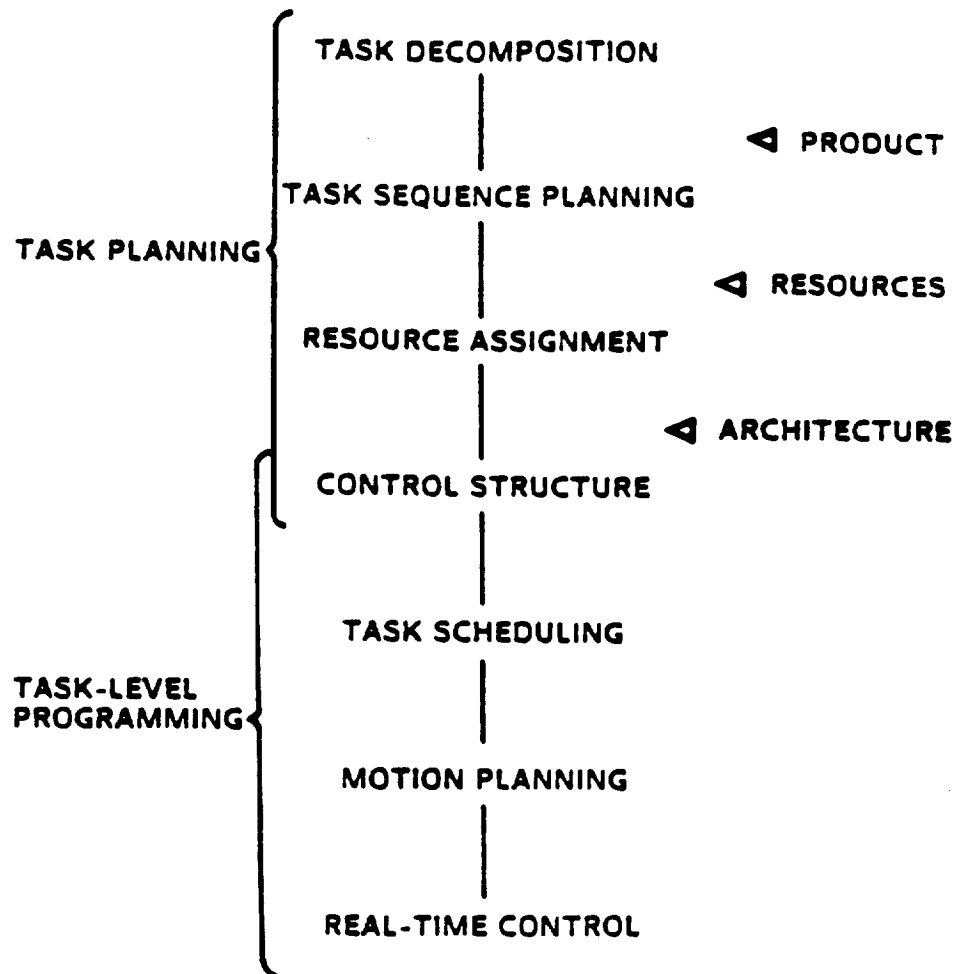
**FIGURE 1. Flexible Assembly Workstation Developed for Electronics Manufacturing (Reprinted from [2]).**

The planning and programming that are required to design and implement an assembly work cell are usually organized into a hierarchical set of levels such as that in Figure 2. Many similar hierarchies have been described in the literature, and we will not attempt to discuss differences among these models here. The highest level of implementation involves the planning of the task itself and this is directly related to a representation or description of the product and its parts. The decomposition of the task must then be coordinated with the available set of resources such as robots, fixtures, and sensors. This decomposed set of tasks is mapped onto a control architecture that defines the coordination and sequencing relationships among the various devices. The scheduling of discrete operations, of robot motions, and the continuous real-time motion control itself are implemented at the lower levels of the structure.

Much of the implementation cost in developing such a system is in the definition of an architecture such that communication between levels remains consistent and reliable, and the implementation of new tasks or the redefinition of tasks, can be accomplished with minimal redesign. One important impact of adaptive and learning technologies, such as neural networks, may be an enhanced capability to develop robust hierarchical systems. Adaptive behavior at one level will ease the requirements for specific coordination with other levels. In addition, broader learning capabilities in general provide the capability to define these structures in a more abstract sense so that they could be adaptive or re-programmed more easily for changing task requirements. In practical industrial situations a large fraction of the implementation costs may be spent on development and programming rather than the capital cost of equipment. The capacity to provide more efficient implementation through automatic learning systems could have significant impact on the economics of building automation systems for manufacturing applications.

Most planning and programming tasks for industrial applications are currently carried out manually. In many cases, the product design, the manufacturing systems plan, and the final manufacturing systems implementation may be carried out by different organizations. The evolution of improved tools and methods to carry out these processes will have an important impact on the effectiveness of manufacturing organizations to respond to new technical and economic opportunities. A key to the development of such improved tools and environments will be the incorporation of both generic methods for computation and reasoning with applications specific knowledge and representation of tasks. The successful utilization of neural networks techniques in the planning and control of manufacturing systems will depend upon the detailed domain specific understanding of the applications area at hand. The demonstration of effective neural network approaches to task planning, sequencing, scheduling, routing, discrete control, sensor based control, fine motion control, or error recovery for a given task domain such as assembly or machining would represent a significant achievement and would emphasize the importance of these computational approaches.

The development of a demonstration of computational performance for a domain specific problem such as assembly or machining requires careful attention to the issue of task representation, including assumptions and constraints which are inherent to that manufacturing domain. In our work on assembly sequence planning [3, 4, 5], we have developed a relational model of product parts geometry and relationships which enable us to reason about the feasibility of task operations and, therefore, successfully generate and evaluate alternative feasible sequences for accomplishing the assembly goals. In the assembly problem, the task decomposes into a sequence of subassembly mating operations, each of which is governed by geometric and mechanical constraints. In an other task domain such as machining, the task may decompose into a sequence of alternative milling or cutting operations, each of which also has its own geometric and mechanical constraints. In each of these problem domains, the search over alternative sequences of operations is closely coupled to the evaluation of feasibility predicates which incorporate geometrical and physical reasoning problems. A new approach to the

**ASSEMBLY SYSTEM DESIGN**

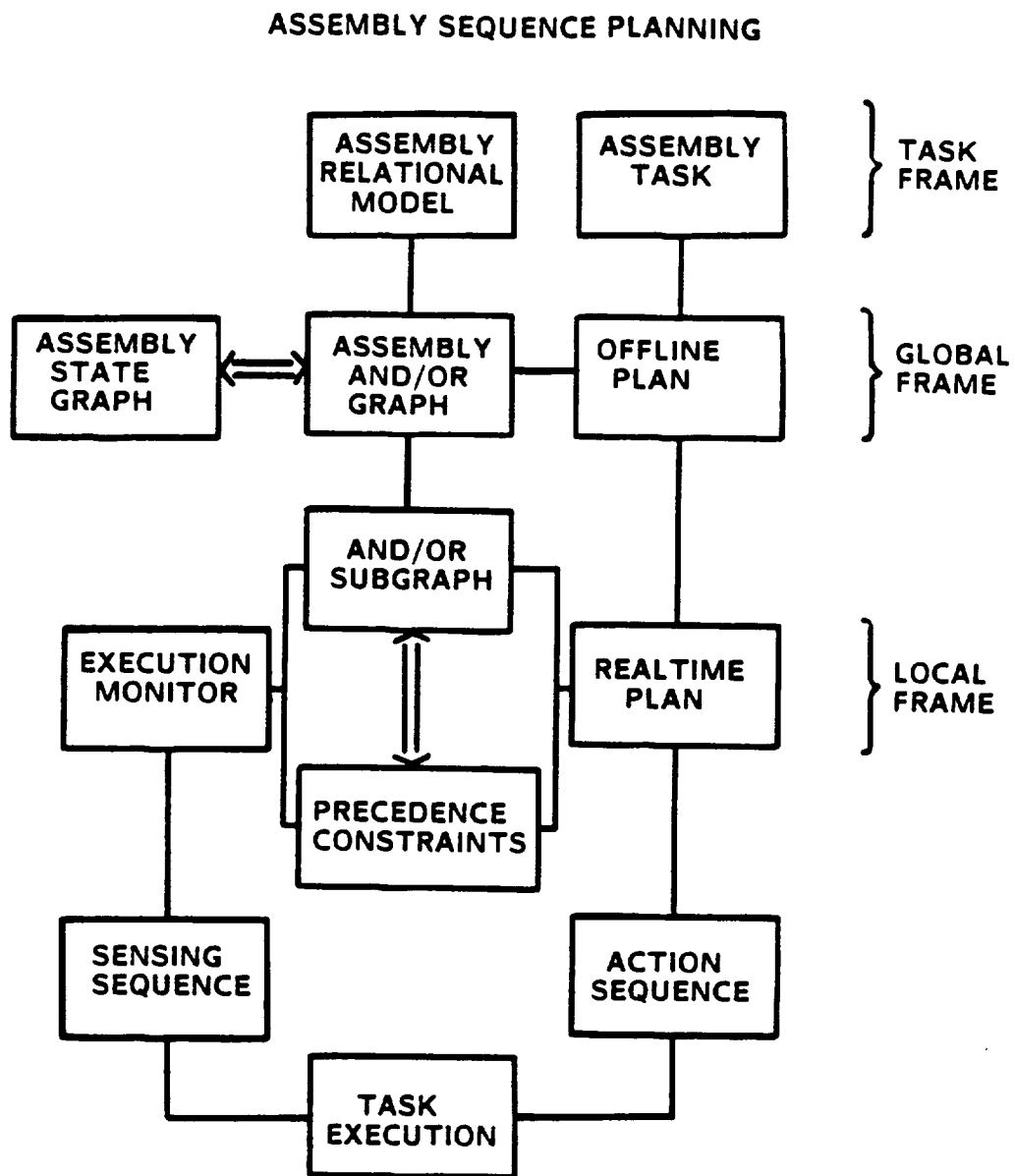
**FIGURE 2. Hierarchical Approach to Assembly Systems Design.**

representation of the geometric or physical relationships appropriate to these elementary operations which lends itself to efficient computation using neural networks would be a very important contribution to the development of these design and planning tools.

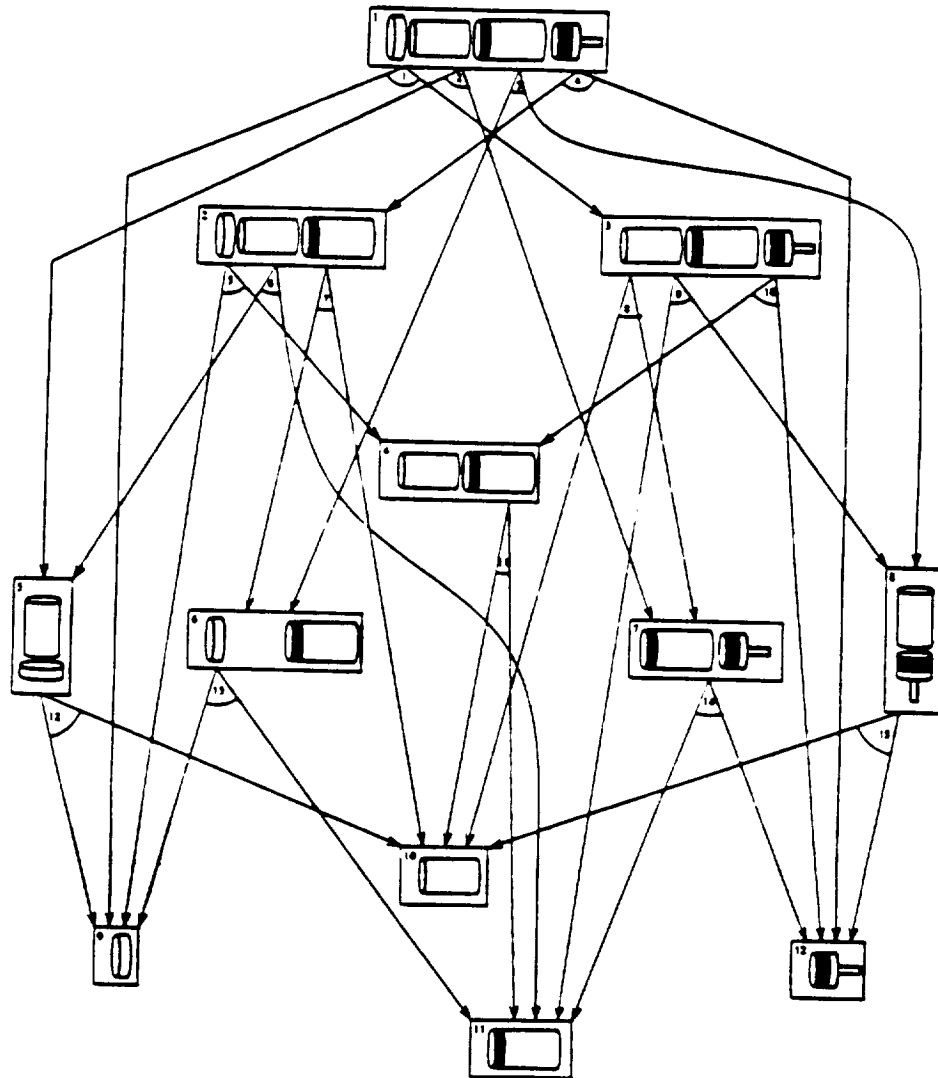
An appropriate domain specific approach may also significantly simplify the representation of the task and lend itself to more efficient planning of sequences. In our approach [3, 4, 5], we have introduced the AND/OR graph structure as a representation of feasible assembly plans. Such an AND/OR graph representation is a distributed state representation for the assembly or disassembly process. In our work on assembly sequence planning, we have demonstrated the completeness and correctness of this AND/OR graph representation, and have shown the equivalence of this representation to a directed graph of assembly states as well as to several classes of precedence relation representations. We have shown that the AND/OR graph representation is more efficient for planning purposes than the directed graph of assembly states. An example of this AND/OR graph representation of assembly plans is shown in Figure 3.

The AND/OR graph plan representation represents alternative assembly sequence plans as tree structures. This same tree structure may be used as a framework for the planning of more detailed operations and motions required in the implementation of the task. Figure 4 shows an example of a tree structure which incorporates the discrete operations associated with devices and mechanisms, including robots and sensors. In this case, the task plan chosen as a single tree from the AND/OR graph has been augmented by the incorporation of the robot and fixture devices. Figure 5 shows the extension of the same framework to the description of continuous motion operations at the lowest level of the control structure. Such a hierarchy between high level symbolic operations sequence planning and low level continuous motion planning is typical of the requirements of a hierarchical control architecture. The use of a common task representation which maps between levels, facilitates the integration of low level functions such as path planning, kinematic learning and control, dynamic learning and control, and sensor recognition which might be approached using neural net computational techniques. Figure 6 shows one example of a control architecture for such a hierarchical system. In this example, the real time motion is executed in conjunction with a real time planning system which modifies the execution of operations according to changing task constraints. Both the real time and off-line planning systems require a common task representation which may be accessed in order to reason about alternative sequences which accomplish the task goals.

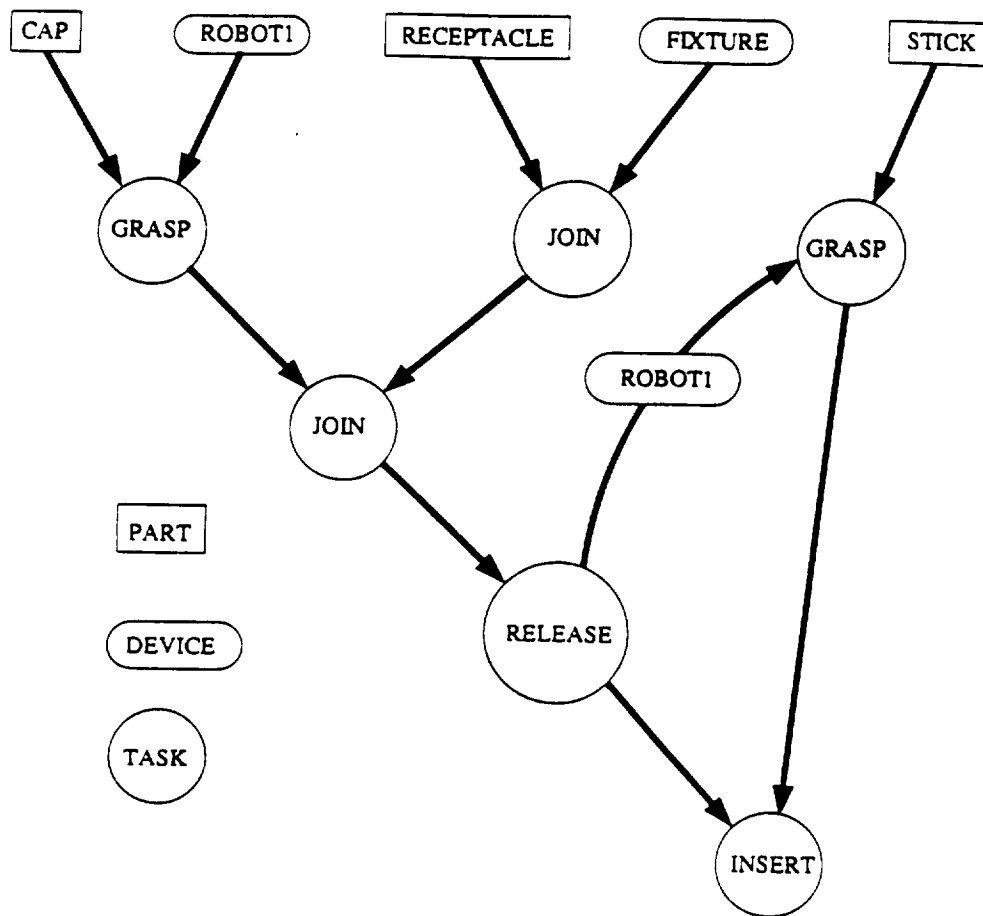
In assembly planning as well as other task planning problems, one cannot explicitly generate all of the feasible plans due to the combinatorial growth in the number of possible sequences. Instead, one must invoke some form of evaluation or objective function in order to choose among feasible sequences and examine in detail only those candidate plans which are most desirable. In assembly planning, these objective functions are related to the complexity of the manipulation operations, the cost of the resources required to execute these operations, the time required for execution, and the complexity and cost of fixtures and tooling used in the implementation of the system. Similar goals and constraints occur in other task planning domains such as metal cutting or parts molding. Such objective functions are extremely difficult to specify in either an explicit and analytical form or even as a heuristic knowledge base. An adaptive or learning system which could synthesize such evaluation functions for a given task domain could facilitate the process of task planning for these applications. Within a hierarchy, a neural net might be used to synthesize the objective function then as a computational approach to minimizing an objective function. This on-line search problem over a distributed representation may be well-suited to a neural net solution.



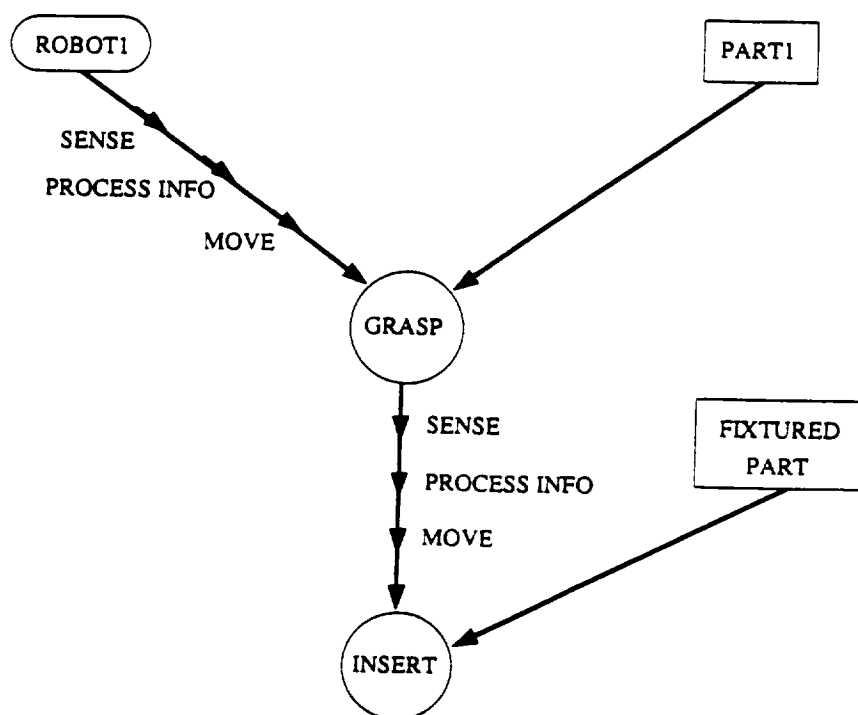
**FIGURE 3. Example of a Control Architecture Which Partitions Off-Line Planning, Real-Time Planning, and Motion Control.**



**FIGURE 4. AND/OR Graph Representation of  
Assembly Plans (See [3, 4, 5]).**



**FIGURE 5. Mapping Discrete Operations Onto the Tree-Structured Assembly Plan**



**FIGURE 6. Mapping Continuous Device Operations Onto the Tree-Structured Assembly Plan.**

### 3. ISSUES AND OPPORTUNITIES

The previous section described an example of the hierarchical system of planning and control which is typical of many manufacturing systems, and suggested ways in which neural net computation might provide an effective tool at the levels of planning, discrete control, continuous control, and sensing. The use of these computational tools will be effective only if they meet needs or expectations of the users. The manufacturer has a number of key practical performance goals which he requires from any system which is being developed. Typically, systems speed, throughput, accuracy, and overall costs of both implementation and operation are factors which he must consider. The flexibility of a system is the ability to change functionality and respond to new requirements, and is an increasingly important component of such systems. The ability to efficiently implement a system, to operate the system reliably, and provide a degree of flexibility which permits an evolution of the manufacturing system in accord with product changes, are important elements which influence the effectiveness of automation in manufacturing today.

Table 1 summarizes a set of technical issues and opportunities which must be addressed in order to expand the capabilities of robotics and automation technology in the manufacturing domain. These issues are grouped into four separate areas: mechanisms, control, representation and planning, and architecture and implementation. While the mechanisms themselves are not directly related to the implementation of adaptive and learning systems, it is clear that improvements in sensing technology, motor technology, and new mechanisms such as flexible arms and sophisticated hands, will place increasingly strong demands on the corresponding control and planning systems to incorporate adaptive capabilities for utilization in specific tasks. There are important opportunities in the development of more robust controllers by utilizing learning systems to more accurately identify robot kinematics and dynamics, to more efficiently adapt dynamic control parameters to particular tasks, and to more effectively integrate sensory information into the control process. The capability to adapt to an inherently uncertain representation or model of the task, is key to the improved reliability of these systems. An automated system for learning of accurate kinematic or dynamic parameters of current robot arms, would have immediate impact in terms of the potential performance of these arms. Newer systems which incorporate light weight, flexible arms or multiarm interactions, will also require such on-line identification in order to function effectively.

Planning and control of the system depends explicitly on the nature of the representation of the robot, task, and environment. In part, a representation is provided by an initial model or description of the system, but increasingly this representation must be updated or derived from sensory information. Utilization of sensory information for the identification of models and for the choice of plans and parameters, are ideal candidates for the application of adaptive and learning systems. A robot, which in response to changing conditions, can adapt its model of the task, the parameters of its control, or the sequence of operations, will provide an important capability for increasingly sophisticated applications. The integration of sensory information from a variety of sources such as vision sensors, tactile sensors, and range sensors has been very difficult to achieve in a purely analytical approach. Multisensor integration through adaptive and learning techniques may be another important opportunity. As discussed previously, the architecture of these systems is typically hierarchical, and the effective integration and coordination among layers of this hierarchy is facilitated by the adaptation of specific functions at one layer in response to generalized commands from a higher level. The basic architectural structure would be retained and the tuning of the coordination and integration parameters would be left to the adaptation mechanism rather than the painstaking trial and error process which is currently employed.

## **TECHNICAL ISSUES IN ROBOTICS AND AUTOMATION FOR MANUFACTURING**

### **MECHANISMS**

- Motor Technologies
- Sensors - Vision, Tactile, Force, Proximity
- Light Weight, Flexible Arms
- Redundant Arms
- Grasping and Hand Design

### **CONTROL**

- Sensor-Based Control/Fine Motion Control
- Adaptive Control/Learning Control
- Flexible Arm
- Multi-Arm
- Dexterous Manipulation

### **REPRESENTATION AND PLANNING**

- Representation of Uncertainty
- Task Planning
- Fine Motion Planning
- Systems Scheduling
- Multi Sensor Integration

### **ARCHITECTURE AND IMPLEMENTATION**

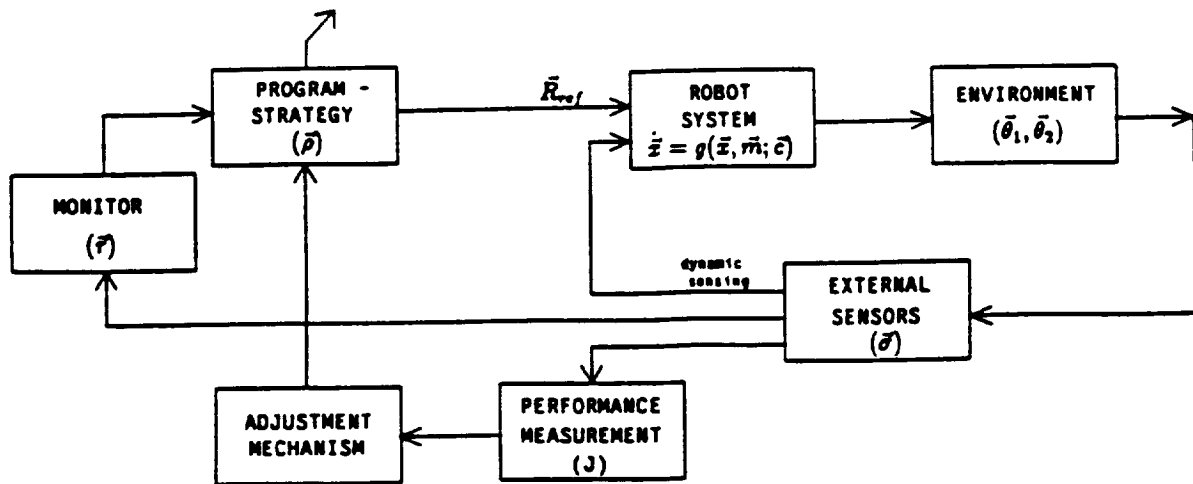
- Hierarchical Architectures
- Product and Systems Design Tools
- User and Programmer Interface
- Programming Language

**TABLE I**

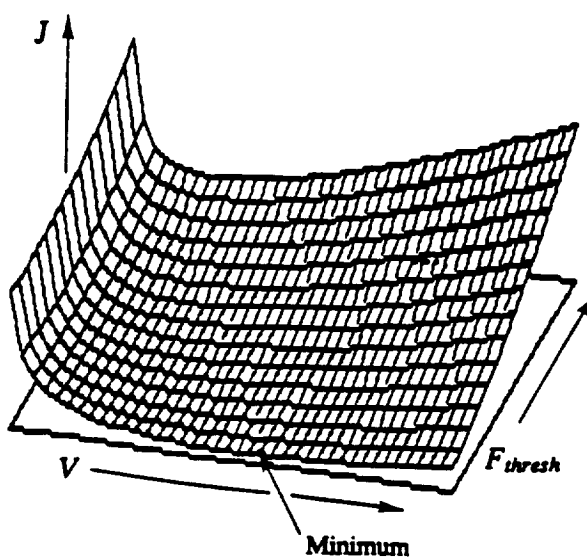
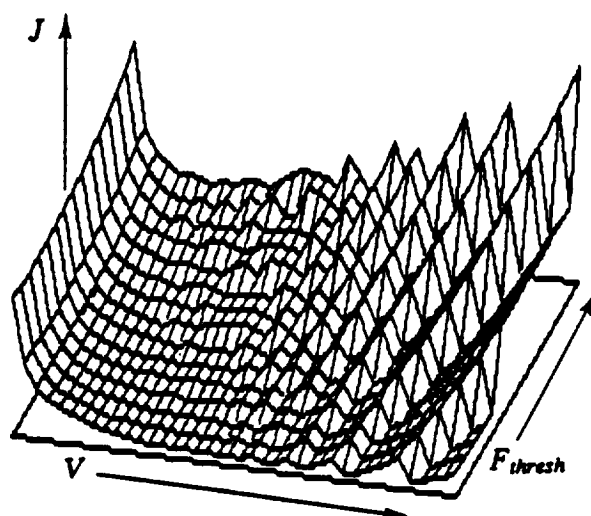
A learning approach which we have used to facilitate the implementation of a robotic system [6], provides an example of parameter learning at the robot operation level. This approach utilizes parameter adaptation within a task which was specified as a set of discrete operations in a conventional high level robot programming language. The structure of this system is shown in Figure 7. The robot automatically generates its own trial and error procedure by modifying the parameters of its program. It generates a representation of the evaluation function parameter space, smooths that optimization surface based on an analytical model of the timing and sampling behavior of the robot itself, and then employs that optimization surface in order to modify the parameters of execution of the robot program in real-time. As the task is repeatedly executed, the evaluation function surface itself is also updated. One example of such a parameter learning task used in this study was the mating of mechanical connectors. In this example the two variable robot program parameters were the velocity of the robot hand and the threshold force for detection of insertion. The performance function  $J$  was the time required to accomplish the task. The resulting performance surface for this robotic insertion task is shown in Figure 8. Notice that the initial surface in the top figure is strongly effected by the sampling times and the instruction execution times of the robot hand. The lower part of Figure 8 shows the smooth performance surface used in the learning procedure. Given this task, the robot performed its own set of trial and error procedures in order to establish the performance surface. The convergence of the performance with the number of experiments is shown in Figure 9. As the robot continued to perform this task, the optimization surface was appropriately modified in real time. This example shows how a robot could effectively learn its performance space on a given task and then update, in real time, execution parameters in order to improve its own performance.

An example of the application of adaptation to a sensor-based control problem is the use of sensor-based control in conjunction with partial state information. In [7], we have described some experiments in which feature information derived from images is used as a partial state representation of the relative position of a robot, and this feature information was directly coupled into a robot joint control loop. Such a hierarchical structure illustrated in Figure 10 utilizes a lower level feature-based control with partial state information, and a higher level position-based control which utilizes a full position interpretation of information from the images. This decomposition of the problem is effective because it matches the dynamic capabilities of the vision system with the speed requirements for real-time robot control. The full image interpretation carried out at a lower speed, sufficient to maintain the stability and reliability of the motion but delegating the real-time sampling and control to the feature based loop. Adaptation is required in this system because an accurate model of the correspondence between feature space and joint space is never in fact available and must be learned or identified in real-time. The examples described in [7] used a classic model reference adaptive control system which implicitly identifies the feature-to-joint sensitivity matrix. The on-line identification of such a nonlinear mapping might be effectively implemented using neural network techniques.

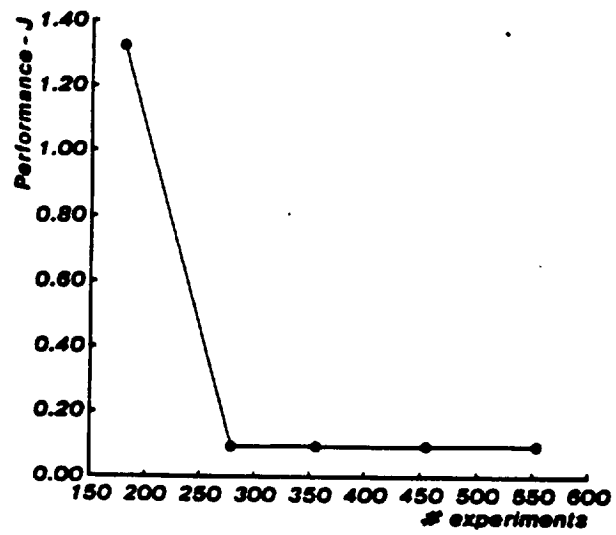
The choice of evaluation functions is fundamental to the problem of learning and adaptation, and in these examples, arises in model identification, parameter learning, and decision making. While model-fitting evaluation functions such as least squared fits are often used for these problems, they are often not necessarily the most desirable, particularly for complex problems. We have been interested in the application of complexity, or representation measures for such problems. In [8], we describe an approach to model size identification which utilizes a minimal representation size criterion, and effectively trades off between the complexity of the chosen model and the accuracy of its fit to data. The further application of such generalized complexity measures to learning problems using neural networks will be extremely interesting to explore.



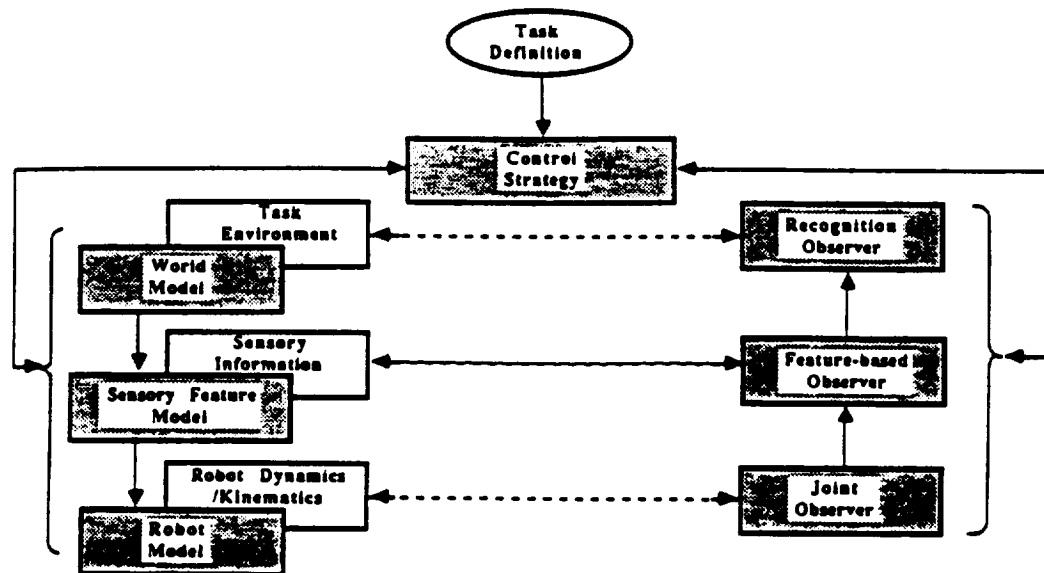
**FIGURE 7** Structure of the Robot Parameter Learning System Described in [6].



**FIGURE 8. Performance Surface Calculated for a Force-Monitored Robot Placement Task [6].**



**FIGURE 9. Convergence of the Parameter Learning Procedure in [6].**



**FIGURE 10.** Structure of a Sensor-Based Control Approach Using Partial-State Information in a Hierarchical System [7].

Several areas where neural networks may play a role in the development of robotics and automation technology for industrial applications are summarized in Table 2. In the short term, a clear opportunity for the impact of learning systems is on the use of sensing and inspection technology for industrial applications. In particular, it would seem that learning systems could offer a marked advantage in the ease of implementation and training of such inspection systems. Developing an inspection tool for new applications is often difficult and expensive, and the ease of developing this system will often be as important as the outright need of the working system. This adaptive or learning capability is of key importance for many short term applications. Another good candidate for such an application in the short term is the area of kinematic calibration of robot arms, utilizing sensing systems to measure positions of the arm end effector. A learning system might identify a complex nonlinear model of the robot arm kinematics which could be used to improve the positioning accuracy of the robot arm itself. A third example which is feasible in the short term, would be the type of parameter learning within the structure of an existing robot program which was illustrated earlier. In this case, the parameter adaptation is essentially a smooth adaptation to local changes and might be handled efficiently by existing neural network techniques. Kinematic path planning is another area of promise, but is in general more difficult because of the dimensionality of the geometric representations required, and the resulting complexity of a neural net implementation.

In the longer term, there are many opportunities for the application of learning systems to task planning and task reasoning problems, particularly those that confront the issue of uncertainty in the task environment. Current experiments in learning of robot dynamics parameters suggest that this is another promising area, and certainly the integration of sensory information into an adaptive robot control structure will be an important element of future robot systems. This will require adaptive systems which both converge quickly, maintain stability, and handle the growing dimensionality of the problem. The use of learning systems to improve the capabilities of planning and execution of fine motion operations such as detailed force control and grasping are promising areas.

An important element in the development of these adaptive and learning techniques and in their evaluation, will be the recognition that robotics is experimental in nature. Development often requires the building of systems and the testing of new tools on real systems in order to evaluate their effectiveness. In applications areas such as manufacturing, this experimental demonstration becomes even more critical since the functional capability of these tools is related most to their ability to compensate for characteristics which are not entirely predictable or which cannot be modeled.

Neural network computing systems with capabilities for supervised learning, matching, and generalization are being developed and explored in a variety of simulated and experimental contexts. Robotic systems offer a promising domain for this exploration since the practical application of complex robotic systems may require adaptive and a learning behavior in order to achieve their desired functionality. In manufacturing, these capabilities in particular, may improve the implementation efficiency, increase the reliability of the system, and improve the performance and accuracy of inspection and control functions. The hierarchical nature of a manufacturing systems architecture lends itself to the integration of these techniques into real systems, the use of neural network techniques in off-line planning, systems design, and product design is an area of particular promise. Neural network principles need to be better understood, and convergence, computational efficiency, and stability characterized more completely. Robotics and automation provide an opportunity for evaluation of these capabilities, and a setting for the development of practical tools to enhance the functionality of robotic systems in manufacturing applications.

## ROLE OF NEURAL NETWORKS IN ROBOTICS FOR MANUFACTURING

### GENERAL:

Autoassociation/Matching  
Classification  
Generalization  
Learning

### SHORT TERM:

Sensor-Based Inspection - Easily Trained  
Sensor Interpretation and Abstraction  
Calibration/Identification  
Parameter Learning in Stereotyped Tasks  
Kinematic Path Planning

### LONG TERM:

Task Planning  
Task Learning in Complex Systems  
Reasoning with Uncertainty  
Learning in Dynamic Control

### OPPORTUNITY:

- Improve Reliability Through Adaptive Behavior
- Improve Implementation Tools Through Trainable Systems

### CAUTION:

- **DON'T IGNORE EXPERIMENTAL APPROACH IN ROBOTICS -  
REAL SYSTEMS**

**TABLE 2**

## ACKNOWLEDGEMENTS

Support for the preparation of this chapter was provided by the New York State Center for Advanced Technology in Automation and Robotics and by the NASA Center for Engineering Excellence at Rensselaer. The author would like to thank Ms. R. Laviolette for assistance in preparation of the manuscript.

## REFERENCES

- [1] Fu, K. S., R. Gonzalez, and G. Lee, (1987). *Robotics: Control, Sensing, and Intelligence*, New York, McGraw Hill.
- [2] Sanderson, A. and G. Perry (1983). Sensor-based Robotic assembly Systems: Research and Applications in Electronics Manufacturing, *Proceedings of the IEEE*, 71, (pp. 856-871).
- [3] Homem de Mello, L. and A. Sanderson (1986). AND/OR Graph Representation of Assembly Plans, Fifth AASI Conference Proceedings, Morgan Kaufman, (pp. 1113-1119).
- [4] Sanderson, A. and L. Homem de Mello (1987). Task Planning and Control Synthesis for Flexible Assembly Systems. In A. K. C. Wong and A. Pugh (Eds.), *Machine Intelligence and Knowledge Engineering for Robotic Applications*, Berlin: Springer Verlag, (pp. 331-353).
- [5] Homem de Mello, L. and A. Sanderson (1988). Automatic Generation of Mechanical Assembly Sequences, *IEEE Transactions on Robotics and Automation* (in press).
- [6] Weiss, L., D. Simon, and A. Sanderson (1987). Self-Tuning of Robot Program Parameters, Proceedings Fifth Yale Workshop on Applications of Adaptive Systems Theory.
- [7] Weiss, L., A. Sanderson, and C. Neuman (1987). Dynamic Sensor-based Control of Robots with Visual Feedback, *IEEE Journal of Robotics and Automation*, RA-3, (pp. 404-417).
- [8] Segen, J. and A. Sanderson (1982). Model Inference and Pattern Discovery by Minimal Representation Methods, Robotics Institute, Carnegie-Mellon University, Technical Report CMU-RI-TR-82-2.